REFERENCES 85

[112] Y. Fang, C. Huang, L. Liu, and M. Xue, "Research on malicious javascript detection technology based on lstm," *IEEE Access*, vol. 6, pp. 59118–59125, 2018.

- [113] C. Fred, "Computer viruses," in *Proceedings of the 7th DoD/NBS Computer Security Conference 1984*, pp. 240–263, 1986.
- [114] R. L. Castro, C. Schmitt, and G. D. Rodosek, "Armed: How automatic malware modifications can evade static detection?," in 2019 5th International Conference on Information Management (ICIM), pp. 20–27, 2019.
- [115] R. L. Castro, C. Schmitt, and G. Dreo, "Aimed: Evolving malware with genetic programming to evade detection," in 2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), pp. 240–247, IEEE, 2019.
- [116] H. Aghakhani, F. Gritti, F. Mecca, M. Lindorfer, S. Ortolani, D. Balzarotti, G. Vigna, and C. Kruegel, "When malware is packin' heat; limits of machine learning classifiers based on static analysis features," in *Proc. of NDSS*, 2020.
- [117] M. Chua and V. Balachandran, "Effectiveness of android obfuscation on evading anti-malware," in *Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy*, CODASPY '18, Association for Computing Machinery, 2018.
- [118] P. Dasgupta and Z. Osman, "A Comparison of State-of-the-Art Techniques for Generating Adversarial Malware Binaries," arXiv e-prints, Nov. 2021.
- [119] G. Lu and S. K. Debray, "Weaknesses in defenses against web-borne malware (short paper)," in *Detection of Intrusions and Malware, and Vulnerability Assessment 10th International Conference, DIMVA. Proceedings* (K. Rieck, P. Stewin, and J. Seifert, eds.), Lecture Notes in Computer Science, 2013.
- [120] M. Payer, "Embracing the new threat: Towards automatically self-diversifying malware,"
- [121] N. Loose, F. Mächtle, C. Pott, V. Bezsmertnyi, and T. Eisenbarth, "Madvex: Instrumentation-based Adversarial Attacks on Machine Learning Malware Detection," arXiv e-prints, p. arXiv:2305.02559, May 2023.
- [122] R. Sasnauskas, Y. Chen, P. Collingbourne, J. Ketema, G. Lup, J. Taneja, and J. Regehr, "Souper: A Synthesizing Superoptimizer," arXiv preprint 1711.04422, 2017.
- [123] J. Cabrera Arteaga, P. Laperdrix, M. Monperrus, and B. Baudry, "Multi-Variant Execution at the Edge," arXiv e-prints, p. arXiv:2108.08125, Aug. 2021.

86 REFERENCES

[124] J. Lettner, D. Song, T. Park, P. Larsen, S. Volckaert, and M. Franz, "Partisan: fast and flexible sanitization via run-time partitioning," in *International Symposium on Research in Attacks, Intrusions, and Defenses*, pp. 403–422, Springer, 2018.

- [125] B. G. Ryder, "Constructing the call graph of a program," *IEEE Transactions on Software Engineering*, no. 3, pp. 216–226, 1979.
- [126] S. Narayan, C. Disselkoen, D. Moghimi, S. Cauligi, E. Johnson, Z. Gang, A. Vahldiek-Oberwagner, R. Sahita, H. Shacham, D. Tullsen, et al., "Swivel: Hardening webassembly against spectre," in USENIX Security Symposium, 2021.
- [127] E. Johnson, D. Thien, Y. Alhessi, S. Narayan, F. Brown, S. Lerner, T. McMullen, S. Savage, and D. Stefan, "Sfi safety for native-compiled wasm," NDSS. Internet Society, 2021.
- [128] J. Cabrera-Arteaga, N. Fitzgerald, M. Monperrus, and B. Baudry, "WASM-MUTATE: Fast and Effective Binary Diversification for WebAssembly," arXiv e-prints, p. arXiv:2309.07638, Sept. 2023.
- [129] M. Willsey, C. Nandi, Y. R. Wang, O. Flatt, Z. Tatlock, and P. Panchekha, "Egg: Fast and extensible equality saturation," *Proc. ACM Program. Lang.*, vol. 5, jan 2021.
- [130] "Stop a wasm compiler bug before it becomes a problem | fastly." https://www.fastly.com/blog/defense-in-depth-stopping-a-wasm-compiler-bug-before-it-became-a-problem, 2021.
- [131] D. Cao, R. Kunkel, C. Nandi, M. Willsey, Z. Tatlock, and N. Polikarpova, "Babble: Learning better abstractions with e-graphs and anti-unification," Proc. ACM Program. Lang., vol. 7, jan 2023.
- [132] R. Tate, M. Stepp, Z. Tatlock, and S. Lerner, "Equality saturation: A new approach to optimization," in *Proceedings of the 36th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '09, (New York, NY, USA), p. 264–276, Association for Computing Machinery, 2009.
- [133] T. D. Morgan and J. W. Morgan, "Web timing attacks made practical," *Black Hat*, 2015.
- [134] T. Schnitzler, K. Kohls, E. Bitsikas, and C. Pöpper, "Hope of delivery: Extracting user locations from mobile instant messengers," in 30th Annual Network and Distributed System Security Symposium, NDSS 2023, San Diego, California, USA, February 27 March 3, 2023, The Internet Society, 2023.